

Amendments to the Claims

Please amend the claims as follows:

1.-11. (Canceled)

12. (Currently Amended) A method for compiling an object-oriented program, the program configured to be executed in a system employing region-based memory-management and the method comprising:
receiving source code for an object-oriented program;
performing a points-to analysis on the source code to develop at least one shape graph for the program;
adding instrumentation to the program, the instrumentation configured:
to cause objects to be created in regions based on information in the shape graph;
and
to cause deletion of all objects in a region when a determination is made that no objects in the region are referenced by any fields outside the region;
compiling the program to produce executable code for the program; and
storing the executable code in a computer-readable storage medium;
~~The method of claim 11,~~ wherein performing a points-to analysis comprises:
creating alias sets for parameters based on statements contained in the methods of the object-oriented program;
unifying the alias sets based on statements contained in the methods of the program;
creating at least one shape graph with nodes defined by alias sets and edges defined by field mappings between alias sets; and
associating the nodes of the shape graph with potential memory regions.

13. (Currently Amended) A method for compiling an object-oriented program, the program configured to be executed in a system employing region-based memory-management and the method comprising:

receiving source code for an object-oriented program;
performing a points-to analysis on the source code to develop at least one shape graph for the program;
adding instrumentation to the program, the instrumentation configured:
to cause objects to be created in regions based on information in the shape graph;
and
to cause deletion of all objects in a region when a determination is made that no objects in the region are referenced by any fields outside the region;
compiling the program to produce executable code for the program; and
storing the executable code in a computer-readable storage medium;
~~The method of claim 11,~~ wherein the added instrumentation at least in part comprises:
region-creation code which creates a region given a shape-graph;
object-allocation code which, given a region and object information, allocates an object within certain region; and
region-lookup code which, given a region and an identifier of a field, identifies the region referenced by that field.

14. (Original) The method of claim 13, wherein the added instrumentation further comprises field-setting code which, given two regions and a reference to a field from an object in one region which references an object in the other region, sets the shape graph to connect the nodes corresponding to the two regions through an edge corresponding to the field.

15. (Original) The method of claim 13, wherein employing region-based memory management comprises at least in part:

keeping a count for each region or set of regions of the number of references made to the objects contained in that region or set; and

upon determining that the count for a region or set of regions is zero, deleting the region or set; and

wherein the added instrumentation further comprises:

incrementing code which, given a region or set of regions, increases the count kept for that region or set; and

decrementing code which, given a region or set of regions, decreases the count kept for that region or set.

16. (Canceled)

17. (Currently Amended) A computer-readable storage medium containing instructions, which, when executed, cause a computer to compile an object-oriented program to be executed in a system employing region-based memory management by performing the following process:

receiving source code for an object-oriented program;

performing a points-to analysis on the source code to develop at least one shape-graph template for the program;

adding instrumentation to the program, the instrumentation configured:

to cause objects to be created in regions based on the shape-graph template; and

to cause deletion of all objects in a region when a determination is made that no objects in the region are referenced by any objects outside the region; and

compiling the program.

~~The computer-readable medium of claim 16,~~ wherein performing a points-to analysis comprises:

creating alias sets for parameters based on statements contained in the methods of the object-oriented program;

unifying the alias sets based on statements contained in the methods of the program;

creating at least one shape graph with nodes defined by alias sets and edges defined by field mapping between alias sets; and

associating the nodes of the shape graph with potential memory regions.

18. (Currently Amended) The computer-readable medium of claim ~~16~~ 17, wherein the added instrumentation at least in part comprises:

region-creation code which creates a region given a shape-graph;
object-allocation code which given a region and object information, allocates an object within certain region; and
region-lookup code which, given a region and an identifier of a field, identifies the region referenced by that field.

19. (Original) The method of claim 18, wherein the added instrumentation further comprises field-setting code which given two regions and a reference to a field from an object in one region which references an object in the other region, sets the shape graph to connect the nodes corresponding to the two regions through an edge corresponding to the field.

20. (Original) The method of claim 18, wherein employing region-based memory management comprises at least in part:

keeping a count for each region or set of regions of the number of references made to the objects contained in that region or set; and

upon determining that the count for a region or set of regions is zero, deleting the region or set; and

wherein the added instrumentation further comprises:

incrementing code which, given a region or set of regions, increases the count kept for that region or set; and

decrementing code which, given a region or set of regions, decreases the count kept for that region or set.

21. (New) The method of claim 12, wherein each region has a shape-graph associated with it and each shape-graph is stored along with the regions it is associated with.

22. (New) The method of claim 13, wherein the region-lookup code operates by:
receiving an identifier of a first region;
receiving an identifier of a field in an object contained in the identified first region which references a target object;

identifying a first node in at least one shape graph which represents the identified first region;

identifying an edge leading from the identified first node which represents the field referencing the target object;

identifying a second node which the identified edge leads to; and

identifying a second region represented by the second node as a region containing the target object.

23. (New) The computer-readable medium of claim 17, wherein each region has a shape-graph associated with it and each shape-graph is stored along with the regions it is associated with.

24. (New) The computer-readable medium of claim 18, wherein the region-lookup code operates by:

receiving an identifier of a first region;

receiving an identifier of a field in an object contained in the identified first region which references a target object;

identifying a first node in at least one shape graph which represents the identified first region;

identifying an edge leading from the identified first node which represents the field referencing the target object;

identifying a second node which the identified edge leads to; and

identifying a second region represented by the second node as a region containing the target object.